

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	845	((345/629).ccls.)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/07 08:50
L2	652	l1 and @ad <= "20010412"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/07 08:51
L3	117	l2 and (display near object)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/07 08:51
L4	5	l3 and (different near2 group)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/07 08:52
S1	444	(display near attribute) and layer	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/11/25 13:23
S2	127	((display near attribute) and layer) and (("345"/\$)!.ccls.)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/03/11 07:56
S3	0	((display near attribute) and layer) and (("345"/\$)!.ccls.) and ((345/606). ccls.)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/03/11 07:56
S4	0	((display near attribute) and layer) and ((382/303).ccls.)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/03/11 07:56
S5	9	((display near attribute) and layer) and (("345"/\$)!.ccls.) and ((345/619). ccls.)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/03/11 07:56

S6	4	((display near attribute) and layer) and (("345/\$)!ccls.) and ((345/581).ccls.)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/03/11 08:21
S7	40	(display and attribute and layer and object) and ((345/619).ccls.)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/03/17 10:05
S8	1	("6091893").PN.	USPAT; USOCR	OR	OFF	2003/03/11 12:08
S9	0	("20020057282A1").PN.	USPAT; USOCR	OR	OFF	2003/03/17 10:05
S10	0	(2002/0057282A1).CCLS.	USPAT; USOCR	OR	OFF	2003/03/17 10:06
S11	0	("us20020057282A1").PN.	USPAT; USOCR	OR	OFF	2003/03/17 10:06
S12	0	("US20020057282A1").PN.	USPAT; USOCR	OR	OFF	2003/03/17 10:06
S13	0	("US20020057282A1").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/03/17 10:07
S14	8985	display and attribute and layer and object	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/03/17 10:13
S15	2782064	yoshida et "al."	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/03/17 10:14
S16	62	Yasunari near yoshida	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/03/17 10:14
S17	2	("6061515").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/11/25 13:25

S18	2	("5903693").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/11/25 13:25
S19	2	("5913205").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/11/25 13:26
S20	2	("5907704").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/11/25 13:28
S21	2	("5831631").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/11/25 13:28
S22	2	("6144962").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/11/25 13:33
S23	2	("6091893").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/11/25 13:36
S24	2	("6031537").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/11/25 13:38
S25	2	("6005578").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/11/25 14:08

S26	2	("5958012").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/11/25 14:09
S27	2	("5317689").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/11/25 14:10
S28	2	("5831618").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/11/25 14:10
S29	518	((345/853).ccls.)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/11/25 14:16
S30	8	((((345/853).ccls.)) and (display near attribute) and layers	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/11/25 14:46
S31	19	((("345"/\$)!ccls.) and (display near attribute) and (object near4 layer)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/11/25 14:46
S32	0	((("345"/\$)!ccls.) and (display near attribute) and (architectual with view)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/11/25 14:48
S33	13	((("345"/\$)!ccls.) and (display near attribute) and (architecture with view)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/11/25 14:56
S34	5	((("345"/\$)!ccls.) and "topology map information"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/11/25 14:59

S35	28	network and "topology map information"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/11/25 15:26
S36	2	("5926177").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/11/25 15:03
S37	0	inventory and network and "topology map information"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/11/25 15:27
S38	0	inventory and "topology map information"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/11/25 15:27
S39	13217	inventory and network	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/11/25 15:27
S40	528	(inventory and network) and topology and map	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/11/25 15:28
S41	0	((inventory and network) and topology and map) and (display near attribut)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/11/25 15:28
S42	0	(inventory and network) and (display near attribut)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/11/25 15:28
S43	3	(inventory and network) and attribut	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/11/25 15:28
S44	2	("5831618").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/11/26 08:33

S45	2	("20020149602").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/05/04 13:09
S46	0	("(displaynearattribute)").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/06 15:40
S47	3609	(display near attribute)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 15:54
S48	26	S47 near5 layer	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 15:41
S49	20	S48 and @ad <= "20010412"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/07 08:50
S50	849	((345/619).ccls.)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/07 08:50
S51	19	S50 and distinguishable	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 16:06
S52	9857	match\$5 and grouping and management	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 16:01

S53	9	S52 and S50	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 16:01
S54	7	S53 and @ad <= "20010412"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 16:05
S55	22055	network near management	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 16:05
S56	13364	S55 and @ad <= "20010412"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 16:06
S57	141	S56 and distinguishable	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 16:07
S58	1	S57 and (("345"/\$)!.ccls.)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 16:09
S59	59	S56 and (("345"/\$)!.ccls.)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 16:10
S60	0	S59 and graphically	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 16:10

S61	49	S59 and graphic\$4	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 16:10
S62	47	S61 and displaying	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 16:11
S63	15	S62 and layout	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 16:11
S64	6	S63 and recogniz\$5	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 16:15
S65	8	S63 and monitoring	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/01/06 16:16



US Patent & Trademark Office

[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide

THE ACM DIGITAL LIBRARY


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)
Terms used **categorizing hardware** and **software graphically**

Found 33,076 of 148,162

Sort results by

Display results

☒ [Save results to a Binder](#)
☒ [Search Tips](#)
☐ Open results in a new window

[Try an Advanced Search](#)
[Try this search in The ACM Guide](#)
Results 41 - 60 of 200 Result page: [previous](#) [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale ☐ ☐ ☐ ☐ ☐

41 [Reprint: Reflections on NoteCards: seven issues for the next generation of hypermedia systems](#)

Frank G. Halasz

August 2001 **ACM Journal of Computer Documentation (JCD)**, Volume 25 Issue 3Full text available: [pdf\(1.68 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

NoteCards, developed by a team at Xerox PARC, was designed to support the task of transforming a chaotic collection of unrelated thoughts into an integrated, orderly interpretation of ideas and their interconnections. This article presents NoteCards as a foil against which to explore some of the major limitations of the current generation of hypermedia systems, and characterizes the issues that must be addressed in designing the next generation systems.

Keywords: Collaborative work, hypermedia, hypertext

42 [Software product lines: organizational alternatives](#)

Jan Bosch

July 2001 **Proceedings of the 23rd International Conference on Software Engineering**Full text available: [pdf\(118.42 KB\)](#)
 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

[Publisher Site](#)

Software product lines enjoy increasingly wide adoption in the software industry. Most authors focus on the technical and process aspects and assume an organizational model consisting of a domain engineering unit and several application engineering units. In our cooperation with several software development organizations applying software product line principles, we have identified several other organizational models that are employed as well. In this article, we present a number of organizational models.

43 [Improving Java performance using hardware translation](#)


Ramesh Radhakrishnan, Ravi Bhargava, Lizy K. John

June 2001 **Proceedings of the 15th international conference on Supercomputing**Full text available: [pdf\(254.91 KB\)](#)
 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

State of the art Java Virtual Machines with Just-In-Time (JIT) compilers make use of advanced compiler techniques, run-time profiling and adaptive compilation to improve performance. However, these techniques for alleviating performance bottlenecks are more effective in long running workloads, such as server applications. Short running Java programs, or client workloads, spend a large fraction of their execution time in compilation instead of useful execution when run using JIT compilers. In ...

44 Tolerating memory latency through software-controlled pre-execution in simultaneous multithreading processors

Chi-Keung Luk

May 2001 **ACM SIGARCH Computer Architecture News , Proceedings of the 28th annual international symposium on Computer architecture**, Volume 29 Issue 2Full text available:  pdf(1.11 MB)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Hardly predictable data addresses in many irregular applications have rendered prefetching ineffective. In many cases, the only accurate way to predict these addresses is to directly execute the code that generates them. As multithreaded architectures become increasingly popular, one attractive approach is to use idle threads on these machines to perform pre-execution—essentially a combined act of speculative address generation and prefetching—to accelerate the main thread ...

45 Formal synthesis and code generation of embedded real-time software

Pao-Ann Hsiung

April 2001 **Proceedings of the ninth international symposium on Hardware/software codesign**Full text available:  pdf(456.04 KB)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Due to rapidly increasing system complexity, shortening time-to-market, and growing demand for hard real-time systems, formal methods are becoming indispensable in the synthesis of embedded systems, which must satisfy stringent temporal, memory, and environment constraints. There is a general lack of practical formal methods that can synthesize complex embedded real-time software (ERTS). In this work, a formal method based on *Time Free-Choice Petri Nets* (TFCPN) is proposed for ERTS synthesis ...

Keywords: Petri Nets, code generation, embedded real-time software, scheduling

46 Task-directed software inspection technique: an experiment and case study

Diane Kelly, Terry Shepard


November 2000 **Proceedings of the 2000 conference of the Centre for Advanced Studies on Collaborative research**Full text available:  pdf(136.53 KB)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Research in software inspection has led to the development of inspection techniques focused on providing structure and guidance to the individual inspector, with the goal of improving effectiveness. This paper defines and investigates a new inspection technique, task-directed inspection, specifically developed for inspecting complex computational code, but capable of being applied in other software domains. Students from the Royal Military College of Canada and Queen's University in Kingston, as ...

Keywords: orthogonal defect classification, software inspection, software inspection process, software reading techniques

47 Concurrent garbage collection using hardware-assisted profiling

Timothy H. Heil, James E. Smith

October 2000 **ACM SIGPLAN Notices , Proceedings of the 2nd international symposium on Memory management**, Volume 36 Issue 1Full text available:  pdf(1.74 MB)Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

In the presence of on-chip multithreading, a Virtual Machine (VM) implementation can readily take advantage of *service threads* for enhancing performance by performing tasks such as profile collection and analysis, dynamic optimization, and garbage collection concurrently with program execution. In this context, a hardware-assisted profiling

mechanism is proposed. The *Relational Profiling Architecture* (RPA) is designed from the top down. RPA is based on a relational model similar ...

48 A conceptual investigation of the e-commerce industry

Veda C. Storey, Detmar W. Straub, Kathy A. Stewart, Richard J. Welke
July 2000 **Communications of the ACM**, Volume 43 Issue 7


Full text available:  [pdf\(308.37 KB\)](#)
 [html\(31.07 KB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)



49 Hardware/software synthesis of formal specifications in codesign of embedded systems

Vincenza Carchiolo, Michele Malgeri, Giuseppe Mangioni
July 2000 **ACM Transactions on Design Automation of Electronic Systems (TODAES)**, Volume 5 Issue 3

Full text available:  [pdf\(281.08 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

CoDesign aims to integrate the design techniques of hardware and software. In this work, we present a CoDesign methodology based on a formal approach to embedded system specification. This methodology uses the Templated T-LOTOS language to specify the system during all design phases. Templated T-LOTOS is a formal language based on CCS and CSP models. Using Templated T-LOTOS, a system can be specified by observing the temporal ordering in which the events occur from the outside. In this paper ...

Keywords: codesign, embedded system, hardware and software synthesis

50 Producing more reliable software: mature software engineering process vs. state-of-the-art technology?

James C. Widmaier
June 2000 **Proceedings of the 22nd international conference on Software engineering**

Full text available:  [pdf\(98.31 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

A customer of high assurance software recently sponsored a software engineering experiment in which a real-time software system was developed concurrently by two popular software development methodologies. One company specialized in the state-of-the-practice waterfall method rated at a Capability Maturity Model Level 4. A second developer employed his mathematically based formal method with automatic code generation. As specified in separate contracts, C++ code plus development documentatio ...

Keywords: capability maturity model, formal methods, software engineering experiment, software process and product metrics, software reliability

51 A formal approach for designing CORBA based applications

Matteo Pradella, Matteo Rossi, Dino Mandrioli, Alberto Coen-Porisini
June 2000 **Proceedings of the 22nd international conference on Software engineering**

Full text available:  [pdf\(236.17 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The design of distributed applications in a CORBA based environment can be carried out by means of an incremental approach, which starts from the specification and leads to the high level architectural design. This is done by introducing in the specification all typical elements of CORBA and by providing a methodological support to the designers. The paper discusses a methodology to transform a formal specification written in TRIO into a high level design document written using an extension ...



Keywords: CORBA, design, formal methods, supervision and control system, temporal logic

52 An approach to architectural analysis of product lines

Gerald C. Gannod, Robyn R. Lutz

June 2000 **Proceedings of the 22nd international conference on Software engineering**

Full text available:  pdf(337.95 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

This paper addresses the issue of how to perform architectural analysis on an existing product line architecture. The contribution of the paper is to identify and demonstrate a repeatable product line architecture analysis process. The approach defines a "good" product line architecture in terms of those quality attributes required by the particular product line under development. It then analyzes the architecture against these criteria by both manual and tool-supported methods ...

Keywords: interferometry software, product lines, software architecture analysis, software architecture

53 Investigating and improving a COTS-based software development

M. Morisio, C. B. Seaman, A. T. Parra, V. R. Basili, S. E. Kraft, S. E. Condon

June 2000 **Proceedings of the 22nd international conference on Software engineering**

Full text available:  pdf(129.13 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The work described in this paper is an investigation of COTS-based software development within a particular NASA environment, with an emphasis on the processes used. Fifteen projects using a COTS-based approach were studied and their actual process was documented. This process is evaluated to identify essential differences in comparison to traditional software development. The main differences, and the activities for which projects require more guidance, are requirements definition and COTS ...

Keywords: COTS, commercial off-the-shelf, component-based, empirical study, software development process

54 Understanding the backward slices of performance degrading instructions

Craig B. Zilles, Gurindar S. Sohi

May 2000 **ACM SIGARCH Computer Architecture News , Proceedings of the 27th annual international symposium on Computer architecture**, Volume 28 Issue 2

Full text available:  pdf(128.47 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

For many applications, branch mispredictions and cache misses limit a processor's performance to a level well below its peak instruction throughput. A small fraction of static instructions, whose behavior cannot be anticipated using current branch predictors and caches, contribute a large fraction of such performance degrading events. This paper analyzes the dynamic instruction stream leading up to these performance degrading instructions to identify the operations necessary to exec ...

55 Reflections on designing field research for emerging IS topics: the case of knowledge management

Inger V. Eriksson, G. W. Dickson, Omar A. El Sawy

April 2000 **Communications of the AIS**

Full text available:  pdf(171.48 KB)

Additional Information: [full citation](#), [references](#)

56 Session summaries from the 17th symposium on operating systems principle (SOSP'99)

Jay Lepreau, Eric Eide

April 2000 **ACM SIGOPS Operating Systems Review**, Volume 34 Issue 2

Full text available:  pdf(3.15 MB) Additional Information: [full citation](#), [index terms](#)



57 Distributed query optimization using PERF joins

Ramzi Haraty, Roula Fany

March 2000 **Proceedings of the 2000 ACM symposium on Applied computing**

Full text available:  pdf(406.83 KB) Additional Information: [full citation](#), [references](#), [index terms](#)



Keywords: PERF joins, distributed query optimization, semi joins

58 The friendly intelligent tutoring environment

Ljubomir Jerinic, Vladan Devedzic

January 2000 **ACM SIGCHI Bulletin**, Volume 32 Issue 1

Full text available:  pdf(1.49 MB) Additional Information: [full citation](#), [abstract](#), [index terms](#)


The advancement of using the Artificial Intelligence (AI) methods and techniques in design Intelligent Tutoring Systems (ITSs) makes understanding them more difficult, so that teachers are less and less prepared to accept these systems. As a result, the gap between researchers in the field of ITSs and the educational community is constantly widening. While ITSs are becoming more common and proving to be increasingly effective, each one must still be built from scratch at a significant cost. Also ...



59 Overcoming the challenges to feedback-directed optimization (Keynote Talk)

Michael D. Smith

January 2000 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN workshop on Dynamic and adaptive compilation and optimization**, Volume 35 Issue 7

Full text available:  pdf(1.33 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)


Feedback-directed optimization (FDO) is a general term used to describe any technique that alters a program's execution based on tendencies observed in its present or past runs. This paper reviews the current state of affairs in FDO and discusses the challenges inhibiting further acceptance of these techniques. It also argues that current trends in hardware and software technology have resulted in an execution environment where immutable executables and traditional static optimizations are ...



60 Modeling lot routing software through discrete-event simulation

Chad D. DeJong, Thomas Jefferson

December 1999 **Proceedings of the 31st conference on Winter simulation: Simulation--- a bridge to the future - Volume 1**

Full text available:  pdf(67.84 KB) Additional Information: [full citation](#), [references](#), [index terms](#)



Results 41 - 60 of 200

Result page: [previous](#) [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)